SPECIFICATION AMENDMENTS

None

CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- Before this Amendment: Claims 1-2, 6, 8-13 and 15-32.
- After this Amendment: Claims 1-2, 6, 8-13 and 15-32.

Non-Elected, Canceled, or Withdrawn claims: 3-5, 7 and 14.

Amended claims: 1.

New claims: None.

Claims:

1. (Currently Amended) A system, embedded at least in part on a tangible computer readable medium for enabling interoperability between two graphics technologies, comprising:

a first graphics system configured to render window content in a first mode, the first graphics system being further configured to reference a first type of window using a window handle associated with an instance of the first type of window, wherein rendering of window content is facilitated by a device context that is based on the window handle associated with the instance of the first type of window;

a second graphics system configured to render windows in a second mode without using any device context, the second graphics system being further configured to reference a second type of window without using any window handle; and

an interoperability component configured to cause a dummy window handle to be created for an instance of a window of the second type and to use the dummy window handle if called to perform a graphics related action on the instance of the window of the second type, wherein a null device context is associated with the dummy window handle to facilitate a lookup of the second type of window, wherein any drawing done to the null device context is lost.

2. (Original) The system recited in claim 1, further comprising an application program including a first window and a second window, the first window being of the first type and the second window being of the second type.

3-5. (Canceled)

6. (Previously Presented) The system recited in claim 1, wherein the second graphics system is configured to create a mapping from the dummy window handle to a node in an internal construct used by the second graphics system to manage windows of the second type.

7. (Canceled)

8. (Original) The system recited in claim 1, wherein the second graphics system is further configured to create a render target for receiving rendered window content.

9. (Original) The system recited in claim 8, wherein the render target resides in system memory.

10. (Original) The system recited in claim 8, wherein the render target resides in video memory.

11. (Original) The system recited in claim 8, wherein the render target records rendering commands generated for windows of the second type and that are played back during composition to generate display output.

12. (Previously Presented) A tangible computer-readable storage medium having computer executable components for enabling interoperability between two graphics technologies, comprising:

an interoperability component that interfaces with an application program, the application program including a first window and a second window, the first window being compatible with a first graphics system that uses window handles to reference windows, the second window being compatible with a second graphics system that does not use window handles; and

a mock window handle associated with the second window, the mock window handle indicating that the second window is compatible with the second graphics system, wherein a null device context is associated with the mock window handle to facilitate a lookup of the second window, wherein any drawing done to the null device context is lost.

13. (Previously Presented) The tangible computer-readable storage medium recited in claim 12, further comprising a mapping, maintained by the second graphics system, from the mock window handle to a node in an internal construct used by the second graphics system to manage the second window.

14. (Canceled)

15. (Previously Presented) The tangible computer-readable storage medium recited in claim 12, wherein the second graphics system is further configured to create a render target for receiving rendered window content.

16. (Previously Presented) The tangible computer-readable storage medium recited in claim 15, wherein the render target comprises a software render target.

17. (Previously Presented) The tangible computer-readable storage medium recited in claim 15, wherein the render target comprises a hardware render target.

18. (Previously Presented) The tangible computer-readable storage medium recited in claim 15, wherein the render target records rendering commands generated for the second window and that are played back during composition to generate display output.

19. (Previously Presented) The tangible computer-readable storage medium recited in claim 12, wherein the mock window handle is associated with the null device context associated with the second window to facilitate interoperability with the first graphics system.

20. (Previously Presented) The tangible computer-readable storage medium recited in claim 19, wherein the device context comprises a null device context.

21. (Previously Presented) A computer-implemented method for enabling interoperability between two graphics technologies, comprising:

receiving a request to create a new window;

determining if the new window is of a type associated with an alternative graphics system that does not use a window handle;

if the new window is of a type associated with an alternative graphics system, creating a dummy window handle for the new window to facilitate interoperability with a conventional graphics system;

creating a new visual to be created in connection with the new window, the visual being a construct associated with the alternative graphics system; and

associating the dummy window handle with the new visual by returning a null device context to facilitate a lookup of the new window, wherein any drawing done to the null device context is lost.

22. (Previously Presented) The computer-implemented method recited in claim 21, wherein if the new window is not of the type associated with the alternative graphics system, rendering the window in accordance with the conventional graphics system.

23. (Previously Presented) The computer-implemented method recited in claim 21, further comprising receiving an instruction to render display content to the new window referenced by the dummy window handle, looking up the new visual based on the association between the dummy window handle and the new visual, and rendering the display content to the new visual.

24. (Original) The computer-implemented method recited in claim 23, wherein rendering the display content to the new visual further comprises issuing rendering commands to a render target associated with the new visual.

25. (Original) The computer-implemented method recited in claim 24, wherein the render target comprises a software render target.

26. (Original) The computer-implemented method recited in claim 24, wherein the render target comprises a hardware render target.

27. (Original) The computer-implemented method recited in claim 24, wherein the render target records rendering commands generated for the new window that are played back during composition to generate display output.

28. (Original) A computer-readable medium encoded with computer executable instructions for performing the method of claim 21.

29. (Previously Presented) The system recited in claim 2, wherein the first mode comprises a compositional mode of graphics technology.

30. (Previously Presented) The system recited in claim 2, wherein the second mode comprises an immediate mode of graphics technology.

31. (Previously Presented) The system recited in claim 6, wherein the internal construct comprises a visual tree, and the node comprises a visual.

32. (Previously Presented) The computer-readable medium recited in claim 13, wherein the internal construct comprises a visual tree, and the node comprises a visual.